

【カーソルを極める】

カーソル

SQLでデータベースを操作するには、**対話型**と**埋込型**があります。これまでは、Windowsのコマンドプロンプト等でSQL文を入力し操作する**対話型**を前提にしてきました。

埋込型は、文字通り、JAVAやC言語等のプログラム言語にSQLを埋込んで操作する方法です。通常はSELECT文を実行すると、複数行の結果が返ってきますが、昔のプログラム言語は、複数行を一括に処理することができませんでした。そこで、SELECT文の実行結果であるテーブルの中の1行1行を特定する道具として**カーソル**を使用します。イメージ的にはテーブルの特定の行を指す矢印みたいなものです。この**カーソル**を使用することで、その行からデータを取り出したり。その行を更新したり、又は削除したりすることができます。

なお、現在のプログラム言語では、**カーソル**を使用することはないということであり、あまり深入りしても仕方ないため、基本事項だけをまとめていきます。

カーソルの使用法

そもそも、現在、カーソルが使われなくなった原因は、プログラム言語によってカーソルの実装方法が大きく異なっていたり、データベースによってカーソルの振る舞いが大きく異なっているためです。

そのため、以下に挙げるカーソルの使用法は、その一例です。ただし、DECLARE、OPEN、FETCH、ループ処理及びCLOSEについては、概ね共通して使用すると考えて大丈夫だと思います。

カーソルは機能（道具）であり、カーソル名（ここではEX1）は、テーブル名や属性名等とは全く無関係

DECLARE EX1 CURSOR FOR SELECT 商品名、在庫数、基準在庫数 **FROM** 在庫

カーソルの使用宣言(DECLARE):SELECT文の結果に対して、カーソルの使用を宣言

OPEN EX1

カーソルを開く(OPEN):SELECT文の実行とカーソルの使用を開始

FETCH NEXT FROM EX1 INTO :商品名、:在庫数、:基準在庫数

データの取得(FETCH):SELECT文の結果の中の1行のデータを取得し、3つの変数に代入

WHILE :FETCH_STATUS = 0

BEGIN

UPDATE 納品一覧 **SET** :発注数 = :基準在庫数 - :在庫数

WHERE CURRENT OF EX1

FETCH NEXT FROM EX1 INTO :商品名、:在庫数、:基準在庫数

END

ループ処理(WHILE、BEGIN、END):グローバル変数であるFETCH_STATUSは、直前のFETCHの結果（取得成功は0、失敗は1、取得するデータなしは2）が入る。FETCH_STATUSが0ならば、BEGINとEND間をループする。最後の行をUPDATEした後のFETCHは、FETCH_STATUSが2になり、ループが終了する。

CLOSE EX1

カーソルを閉じる(CLOSE):カーソルの使用を終了

現在のFETCHした行のデータから「基準在庫数 - 在庫数」を求め、その値を納品一覧のは発注数として更新

※上記には「**FETCH NEXT FROM EX1 INTO** :商品名、:在庫数、:基準在庫数」が2回出現するが、**最初**は先頭行のデータを取得して、**2回目**はループ内に出現することによって、先頭行の次の行から最後の行までのデータを1行ずつ取得する。